第 37 卷第 1 期 计算机应用研究 Vol. 37 No. 1
录用定稿 Application Research of Computers Accepted Paper

基于自然最近邻相似图的谱聚类 *

刘友超，张曦煌

(江南大学 物联网工程学院, 江苏 无锡 214122)

摘　要：谱聚类是基于谱图划分理论的一种聚类算法，由于其对非凸数据集具有优越的性能而广受欢迎，但是传统谱聚类算法经常在处理一些结构复杂的数据集时效果不甚理想，并且其相似度矩阵构造时参数的选取往往需要依靠多次实验及个人经验。在这种情况下，提出一种基于自然最近邻相似图的谱聚类（NSG-SC）算法。自然最近邻是一种新颖的最近邻概念，可以有效地避免 K 最近邻以及 ε-最近邻方法需要人为设置参数的缺点。该算法构造相似度矩阵时依靠数据集自身的特性进行搜索，避免了参数选取不当以及离散点所带来的影响，更加真实地反映了数据集的结构关系。实验结果表明，提出的 NSG-SC 算法具有可行性和有效性。

关键词：谱聚类；自然最近邻；相似图；相似度矩阵

中图分类号：TP301.6　　doi: 10.19734/j.issn.1001-3695.2018.06.0460

Spectral clustering based on natural nearest neighbor similarity graph

Liu Youchao, Zhang Xihuang

(*School of Internet of Things Engineering*, *Jiangnan University*, *Wuxi Jiangsu* 214122, *China*)

Abstract: The spectral clustering is a clustering algorithm based on the theory of spectral partitioning, and it is a popular method due to its superior performance in the data sets with non-convex clusters. But the traditional spectral clustering algorithm cannot often get correct results on complex data sets, and the choice of parameters of affinity matrix construction depends on multiple tests and personal experience. Based on the situation, this paper proposes a spectral clustering algorithm based on natural nearest neighbor similarity graph（NSG-SC）. Natural nearest neighbor is a novel concept in terms of nearest neighbor, and it can avoid the disadvantages of K-nearest neighbor and ε-nearest neighbor. They usually need set parameters artificially effectively. The algorithm constructs an affinity matrix depending on the characteristics of the data sets, and it avoids some adverse effects. It is that inappropriate choice of parameters and isolated points cause them. The algorithm can also reflect better characteristics of data. The results of experiments show that the proposed algorithm named NSG-SC has feasibility and effectiveness.

Key words: spectral clustering; natural nearest neighbor; similarity graph; affinity matrix

0　引言

聚类分析是机器学习领域的一个重要分支，是人们认识和探索数据之间内在联系的有效手段。聚类分析作为一种重要的无监督学习方法，其主要思想是按照特定的标准，将数据划分到多个互不相交的簇，使其满足簇内数据具有较高的相似性，而簇间数据具有较低的相似性[1]。

到目前为止，已经有许多聚类的算法被提出。比较典型的有：基于划分的聚类方法，如 K-means[2]、K-medoids[3]等；基于密度的聚类方法，如 DBSCAN（density-based spatial clustering of applications with noise）[4]、OPTICS（ordering points to identify the clustering structure）[5]等；基于网格的聚类方法，如 STING（statistical information grid）[6]等；基于层次的聚类方法，如 ROCK(A hierarchical clustering algorithm for categorical attributes)[7]、CURE（clustering using representatives）[8]等；基于模型的聚类方法以及基于图论的谱聚类方法。

谱聚类算法建立在谱图理论基础之上，其本质是利用谱松弛方法将聚类问题转换为图的最优划分问题。对比传统聚类算法，其能够在任意形状的样本空间上完成聚类，并且收敛于全局最优解。因此谱聚类也被广泛应用于生物信息学[9]，模式识别[10]，图像分割[11]及文本挖掘[12]等领域。

比较经典的谱聚类算法有 Ng 等人提出的 k 路划分的 NJW 谱聚类算法[13]以及 Zelnik-Manor 提出的自适应谱聚类算法[14]等。目前，对于谱聚类的研究主要集中在相似度矩阵构造、特征向量选取、自动确定聚类数目、拉普拉斯矩阵选取和海量数据运用等方面[15]。

收稿日期：2018-06-20；**修回日期：**2018-07-31　**基金项目：**江苏省产学研合作项目（BY2015019-30）

作者简介：刘友超（1994-），男，硕士研究生，主要研究方向为数据挖掘（276237597@qq.com）；张曦煌（1962-），男，教授，博士，主要研究方向为分布式系统与应用.

在对谱聚类已知的这些研究方向中，相似度矩阵的构造是重中之重。因为其直接影响到特征向量的获取，从而影响到最终聚类的结果。有关谱聚类中相似度矩阵构造的研究一直没有停止。2011 年，Yang 等人提出了一种密度敏感的距离度量方法[16]。该度量方法定义了一个可调节长度的线段，该线段能适应不同密度区域的距离度量。在高密度区域中线段缩短，而在低密度区域中该线段则相应地拉长。该算法能处理多尺度聚类问题，对参数选择相对不敏感，但也存在着聚类效果不稳定，真实数据集上的效果欠佳等问题。2012 年，Li 等人使用邻近传播原则提出了一种新的相似度矩阵构建方法[17]，该相似度矩阵能增加同一簇中点对的相似度，从而更好地检测数据结构。2013 年，Blekas 等人提出了一个基于牛顿运动方程的谱聚类算法[18]。他们建立了一个潜在的轨道分析交互模型并且使用了牛顿预处理方法去获得有价值的相似性信息，丰富了相似度矩阵。2015 年，Inkaya 等人提出了一个基于密度和连通性的自适应相似图构建算法[19]，利用该算法可以对数据集构建相似图，之后再由相似图得到相似度矩阵，由此得到了一种新的谱聚类算法[20]。该算法有着能找到具有任意形状和可变密度的簇的局部特征以及表现稳定等优点，但是也存在着对噪声点的处理能力不强和存在混合聚类时数据点之间的邻近关系不是很精确等问题。

自然最近邻[21]是一种新型的最近邻概念，属于无尺度最近邻方法的范畴。该方法依靠数据集自身特性搜索，能有效地避免 K 最近邻以及 ε-最近邻方法需要人为设置参数的缺点。本文借鉴了自然最近邻概念，用以代替传统谱聚类算法构建相似度矩阵时所采用的 K 近邻法、ε 近邻法或全连接法。利用自然最近邻搜索算法构建一个免参数的相似图，再利用高斯核函数根据相似图构建相似度矩阵，最后再结合经典谱聚类算法的步骤，就可以得到一种基于自然最近邻相似图的谱聚类算法（spectral clustering based on natural nearest neighbor similarity graph，简称 NSG-SC）。实验证明，本文所提出的 NSG-SC 算法具有更加优秀的聚类效果。

1　相关研究

1.1　自然最近邻

最近邻的概念早在 1951 年就已经被提出，一经提出便受到广泛的关注及研究，现已广泛应用于人工智能、数据挖掘及模式识别等领域。现在使用最为广泛的两个最近邻概念均由 Stenvens 提出，分别是 K 最近邻及 ε-最近邻。K 最近邻的基本思想是找出数据集中每个对象周围与其距离最短的 K 个对象，其中参数 K 需要人工设置。ε-最近邻的基本思想则是找出数据集中每个对象周围半径 ε 范围内的对象，其中参数 ε 需要人工设置。可以看出，无论是 K 最近邻还是 ε-最近邻，其最近邻的搜索都非常依赖于参数的设置，而不是根据数据集自身的特性进行搜索。

自然最近邻是一种近几年才提出的新型最近邻概念，其属于无尺度最近邻方法的范畴，不需要进行人工的参数设置，这也是其与 K 最近邻和 ε-最近邻最大的区别。该方法受到人类社会友谊关系的启发，可以在不给定参数的情况下，根据数据集自身的属性特点，有效地确定数据集中的邻域，为每个数据点动态地选择数量不同的最近邻点。

自然最近邻的基本思想是根据密度划分，密度较大区域的数据点自然就拥有较多的近邻点，相对地，密度较小区域的数据点拥有的近邻点就较少。而数据集中相对离群的数据点只拥有几个或完全没有近邻点。正因为噪声点和异常点没有近邻点，所以正常点也不会把它们当成近邻点。

定义 1　r 邻域。定义公式如下：

$$KNN_r(x_i)=\bigcup_{n=1}^{r}\{findKNN(x_i,n)\} \tag{1}$$

其中：$findKNN(x_i,n)$ 表示 KNN 搜索函数，它返回 x_i 的第 n 个最近邻，$KNN_r(x_i)$ 表示原始数据集 X 的一个子集。

定义 2　自然最近邻。基于 r 邻域，如果点 X 在点 Y 的 r 邻域中并且点 Y 也在点 X 的 r 邻域中，则称 X 和 Y 互为自然最近邻，具体定义公式如下：

$$\begin{aligned}&x_j\in NN(x_i)\\&\Leftrightarrow(x_i\in KNN_r(x_j))\wedge(x_j\in KNN_r(x_i))\end{aligned} \tag{2}$$

定义 3　稳定搜索状态。当且仅当满足如下条件时，自然最近邻算法达到稳定搜索状态：

$$\begin{aligned}&(\forall x_i)(\exists x_j)(r\in N)\wedge(x_i\neq x_j)\\&\rightarrow(x_i\in KNN_r(x_j))\wedge(x_j\in KNNr(x_i))\end{aligned} \tag{3}$$

其中：r 是搜索轮数，若满足公式(3)则代表该轮搜索是稳定的，不需要提前终止搜索。

算法 1　自然最近邻搜索算法

$Input: the\ data\ set\ X$
$r=1, flag=0, NaN_Edge=\varnothing$
$\forall x_i\in X, NaN_Num(x_i)=0$
$while\ flag==0\ do$
　$for\ all\ x_i\in X\ do$
　　$knn_r(xi)=findKNN(x_i,r)$
　　$KNN_r(x_i)=KNN_r(x_i)\cup\{knn_r(x_i)\}$
　　$if\ x_i\in KNN_r(knn_r(x_i))$
　　　$\&\&\{knn_r(x_i),x_i\}\notin NaN_Edge\ then$
　　　$NaN_Edge=NaN_Edge\cup\{x_i,knn_r(x_i)\}$
　　　$NaN_Num(x_i)=NaN_Num(x_i)+1$
　　　$NaN_Num(knn_r(x_i))$
　　　　$=NaN_Num(knn_r(x_i))+1$
　　$end\ if$
　$end\ for$
　$cnt=count(NaN_Num(x_i)==0)$

　$rep=repeat(cnt)$
　$if\ all(NaN_Num(x_i))\neq 0\ \|\ rep\geq\sqrt{r-rep}\ then$
　　$flag=1$
　$end\ if$
　$r=r+1$
$end\ while$
$\lambda=r-1$
$Output: NaN_Edge$

其中: NaN_Edge 代表的是连接两个顶点之间的边的集合。把数据集 X 中每个数据看做一个顶点，若两个顶点 x_i 和 x_j 之间插入了一条边，则将该边放入 NaN_Edge 集合中。函数 $count$ 的作用是统计 NaN_Num 集合中为 0 的元素的个数。函数 $repeat$ 是为了统计变量 cnt 连续出现的重复次数。

1.2 NJW 谱聚类

NJW 谱聚类算法是 Ng 等人提出的一种比较经典的多路划分谱聚类算法[13]，其构建相似度矩阵采用的是基于高斯核函数的全连接法，该算法的基本步骤如下：

输入：初始数据集 $X=\{x_1,x_2,\cdots,x_n\}$，聚类数 k

输出：聚类结果 $C=\{C_1,C_2,\cdots,C_k\}$

a)构建相似度矩阵 A，定义公式如下：

$$A_{ij}=\begin{cases}e^{-\frac{\|x_i-x_j\|^2}{2\sigma^2}} & if\ \ i\neq j\\ 0 & otherwise\end{cases} \tag{4}$$

其中 σ 为缩放参数，需要手动设置。

b)计算出度矩阵 D 并利用 D 和 A 计算出拉普拉斯矩阵 L。其中度矩阵 D 定义公式如下：

$$D_{ij}=\begin{cases}\sum_{j=1}^{n}A_{ij} & if\ \ i==j\\ 0 & otherwise\end{cases} \tag{5}$$

拉普拉斯矩阵 L 定义公式如下：

$$L=D^{-1/2}AD^{-1/2} \tag{6}$$

c)计算出 L 的前 k 个最大特征向量 $\{z_1,z_2,\cdots,z_k\}$，然后建立矩阵 Z 并将其标准化得到矩阵 Y。其中 Z 定义公式如下：

$$Z=[z_1,z_2,\cdots,z_k]\in R^{n\times k} \tag{7}$$

Y 定义公式如下:

$$Y_{ij}=\frac{Z_{ij}}{\sqrt{\sum_{j=1}^{n}Z_{ij}^2}} \tag{8}$$

2 NSG-SC 算法

2.1 算法思想

本文先根据算法 1 构建数据集的自然最近邻关系集合，得到集合 NaN_Edge。之后再把原数据集中的每个数据点看做一个顶点，得到集合 V。以 NaN_Edge 作为边的关系集合，V 作为顶点的关系集合，由此可以构建一个无向加权图，命名为 NSG (natural nearest neighbor similarity graph)，定义如下：

$$NSG=(V,NaN_Edge) \tag{9}$$

NSG 由许多连通子图组成，其中每个连通子图代表一个潜在的簇，同一连通子图中的任意两个点之间都存在着一条或多条边可以将两点直接或间接连通。确定 NSG 连通子图数量 g，将其与目标聚类数 k 比较，如果 g 大于 k，则表示存在着过多独立的簇。因此，在这种情况下新插入一条边 (v_i,v_j) 在点 x_i 和 x_j 之间。具体的 (v_i,v_j) 定义如下：

$$(v_i,v_j)=\arg\ \min\{d_{ij}\mid v_i\in C_p,v_j\in C_q,p\neq q\} \tag{10}$$

其中 C_p 和 C_q 分别代表一个连通子图。

插入操作完成之后更新连通子图数量 g 和相似图 NSG。重复比较 g 和 k，进行连通子图合并操作直到 g 和 k 相等。NSG 如图 1 所示。其中红色的边代表 NSG 在合并子图之前便存在的边,蓝色的边代表合并子图之后新增加的边。

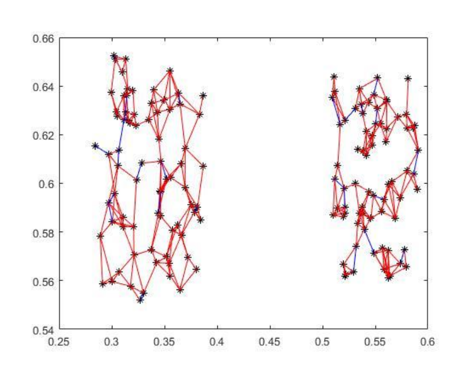


图 1　相似图 NSG

Fig.1　Similarity graph NSG

最后再利用高斯核函数构建相似度矩阵 A，定义公式如下：

$$A_{ij}==\begin{cases}\exp(-\frac{d_{ij}^2}{(\max\{d_{ih}:(i,h)\in \mathrm{NaN_Edge}\})^2}) & \mathrm{if}(i,j)\in \mathrm{NaN_Edge}\\ 0 & \mathrm{otherwise}\end{cases} \tag{11}$$

其中: $\max\{d_{ih}:(i,h)\in NaN_Edge\}$ 表示同一连通子图中最长的边。

2.2 算法描述

输入: 初始数据集 $X=\{x_1,x_2,\cdots,x_n\}$，聚类数 k

输出: 聚类结果 $C=\{C_1,C_2,\cdots,C_k\}$

Step1　根据算法 1 得到关系集合 NaN_Edge

Step2　根据定义式(9)，构建无向加权图 NSG

Step3　确定 NSG 的各个连通子图构成及连通子图数量 g，再通过连通子图互相合并将其数量缩减至 k 个

Step4　利用式(11)构建相似度矩阵 A

Step5　依次进行 1.2 节中 NJW 谱聚类算法的 step2、step3、step4 得到最终聚类结果 C

2.3 算法时间复杂度分析

设待聚类原始数据集 X 的样本数量为 n，根据 1 节算法 1 的步骤描述，在自然最近邻搜索阶段，算法的时间复杂度由以下几个主要步骤决定：a）创建可用于存储数据集的 k-d 树，此步骤的时间复杂度为 $O(n\log n)$；b）对于单独一轮 r 自然最近邻搜索，其时间复杂度为 $O(n\log n)$。一共进行了 λ 轮搜索，所以搜索的总时间复杂度为 $O(\lambda n\log n)$，其中 $2\leq\lambda\leq n$。λ 一般为 6 或 7，对于高维或不规则的数据集，$20\leq\lambda\leq 30$。

根据 2.1 节的 NSG-SC 算法步骤描述，除自然最近邻搜索外其余步骤的时间复杂度由以下几个主要步骤决定：1）构建相似度矩阵，时间复杂度为 $O(n^2)$；2）K-means 步骤，时间复杂度为 $O(nkt)$，其中 t 为迭代次数，一般不超过 300。

综上分析，在 n 较大的情况下，NSG-SC 算法的时间复杂

度仍为 $O(n^2)$，和一般的谱聚类算法时间复杂度相同。

如何确定目标聚类数 k

NSG-SC 算法虽然在相似图构建过程中无须参数，但在之后的步骤中还是需要手动设置参数 k 的值。确定目标聚类数 k 其实是大多数聚类算法都存在的一个普适性问题，目前已经有各式各样或多或少成功的方法为这个问题提供了解决思路。

最常用且简单的方法是可视化数据，之后直接观察出聚成几类比较合适，但通常情况下这种方法并不奏效。在基于模型的聚类中，通常存在比较有效的标准可以从数据中选取 k 值。这个标准通常基于数据的对数似然性，之后可以采取频率或贝叶斯方法来处理[22]。而在对基础模型没有或很少假设的情况下，则一般使用各种不同的指标来选取 k 值。常见地可以使用 ad-hoc 度量方法如簇内和簇间相似性比率，过度信息理论标准[23]和间隔统计量等。

3　实验与分析

3.1　相关算法及参数设置

本文算法分别与 K-means 算法，NJW 谱聚类算法[13]（以下简称 NJW 算法），Self-Tuning 谱聚类算法[14]（以下简称 ST-SC 算法）和文献[20]提出的算法（以下简称 DAN 算法）进行比较。K-means 算法的参数 k 为目标聚类数；NJW 算法的参数 sigma 为缩放参数，在此次实验中选取经验值 sigma=0.005，参数 k 为目标聚类数；ST-SC 算法的参数 K 为构建相似度矩阵时选取的每个点的最近邻 K 个点，这里设置为作者在原文中建议的值 K=7，参数 k 为目标聚类数；DAN 算法的参数 k 为目标聚类数。本文提出的 NSG-SC 算法的参数 k 为目标聚类数。

3.2　人工数据集实验及分析

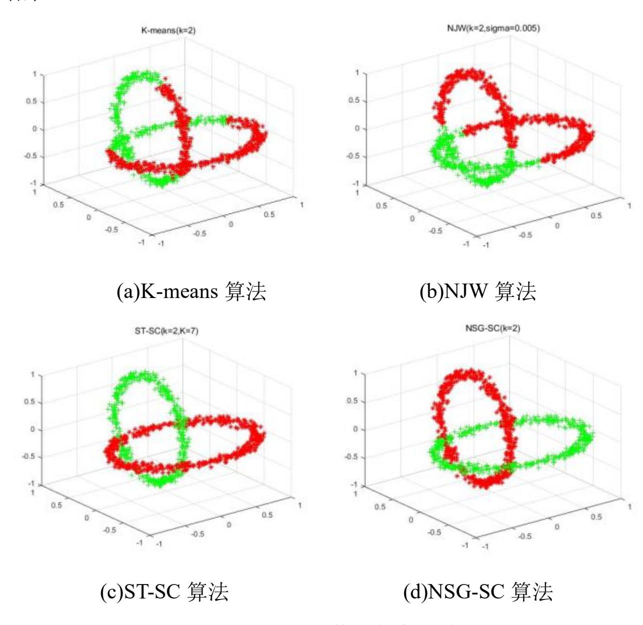
为了验证 NSG-SC 算法的有效性，本文先将 NSG-SC 算法与 K-means 算法、NJW 算法和 ST-SC 算法在图 2 所示的四个人工合成数据集上进行实验。四种人工数据集分别是 ChainLink、Sticks、ThreeCircles 和 UnbalanceSpiral，其详细信息如表 1 所示。分别对这四种人工数据集进行实验后，实验的最终聚类结果如图 2~5 所示。每张图左上为 K-means 算法，右上为 NJW 算法，左下为 ST-SC 算法，右下为 NSG-SC 算法。

表 1　四种人工数据集

Table 1　Four type of artificial data sets

数据集	实例数	维度	类别
ChainLink	1000	3	2
Sticks	512	2	4
ThreeCircles	1801	2	3
UnbalanceSpiral	567	2	3

对比分析可知，本文提出的 NSG-SC 算法在四个数据集上均可以正确聚类；在 ChainLink 数据集上，ST-SC 算法可以正确聚类，K-means 算法和 NJW 算法在两个流形簇相互靠近的地方无法正确聚类；在 Sticks 数据集和 UnbalanceSpiral 数据集上，K-means 算法、NJW 算法和 ST-SC 算法均出现不同程度的错误聚类，这表明簇间较高的相似度会导致这些算法的错误判断；在 ThreeCircles 数据集上，ST-SC 算法可以正确聚类，K-menas 算法和 NJW 算法的聚类则完全错误。通过实验表明，NJW 算法和 ST-SC 算法构建的相似度矩阵在簇间相似度较高的情况下无法真实地反映出数据集结构，从而导致样本错误聚类。而 NSG-SC 算法利用自然最近邻相似图构建出的相似度矩阵，在数据集较为复杂且簇间相似度较高的情况下，仍能正确地反映出数据集的真实结构，从而得到正确的聚类结果。由此分析推断 NSG-SC 算法在处理复杂数据集时往往能获得更优秀的聚类结果。

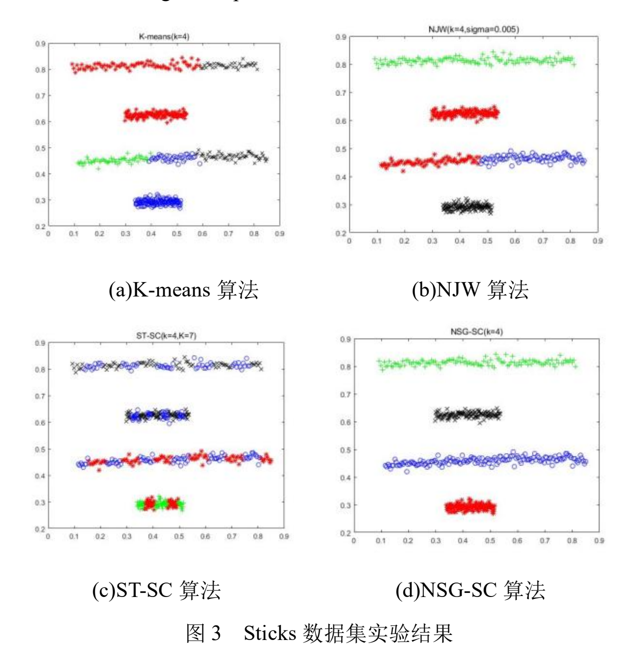


(a)K-means 算法　(b)NJW 算法

(c)ST-SC 算法　(d)NSG-SC 算法

图 2　ChainLink 数据集实验结果

Fig.2　Experiment results of ChainLink



(a)K-means 算法　(b)NJW 算法

(c)ST-SC 算法　(d)NSG-SC 算法

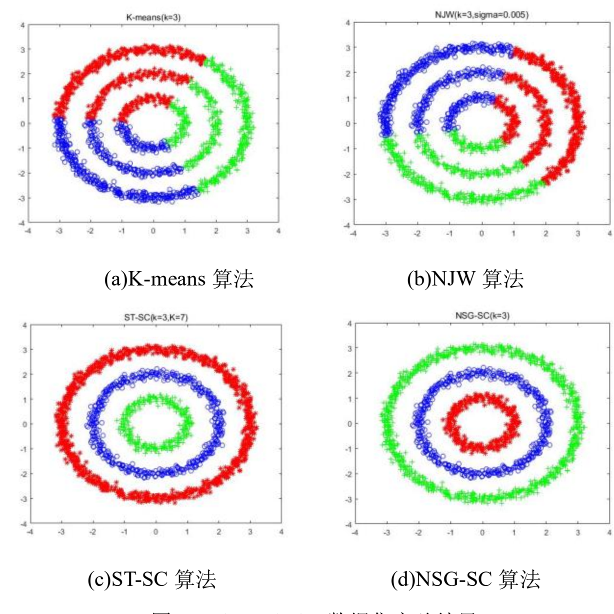
图 3　Sticks 数据集实验结果

Fig.3　Experiment results of Sticks

3.3　评价指标

在接下来的真实数据集实验中，将分别采用 ARI[24]（adjusted rand index）和 AMI[25]（adjusted mutual information）这两个指标来评价 K-means 算法、NJW 算法、ST-SC 算法、

DAN 算法与 NSG-SC 算法的效果。

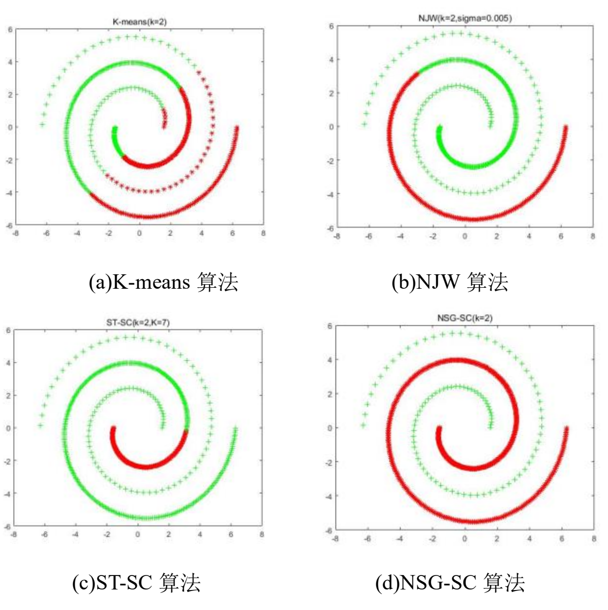


(a)K-means 算法　　(b)NJW 算法

(c)ST-SC 算法　　(d)NSG-SC 算法

图 4　ThreeCircles 数据集实验结果

Fig.4 Experiment results of ThreeCircles



(a)K-means 算法　　(b)NJW 算法

(c)ST-SC 算法　　(d)NSG-SC 算法

图 5　UnbalanceSpiral 数据集实验结果

Fig.5　Experiment results of UnbalanceSpriral

RI（rand index）在统计学中，特别是在聚类中，表示的是两个簇间的相似性度量。从数学的角度来看，RI 指标与准确性有关。RI 指标的取值范围为[0,1]。而 ARI 指标在 RI 指标的基础上进一步实现了“在聚类结果随机产生的情况下，指标应该接近零”的效果。ARI 指标取值范围为[-1,1]，值越大意味着聚类结果与真实情况越吻合。从广义的角度来讲，ARI 指标衡量的是两个数据分布的吻合程度。

MI（mutual information）是信息论里的一种有用信息度量，它可以看成是一个随机变量中包含的关于另一个随机变量的信息量，或者说是一个随机变量由于已知另一个随机变量而减少的不确定性。AMI 指标是对 MI 指标的进一步改进，它常用于聚类，类似于 ARI 指标对于 RI 指标的纠正，并且与信息的变化密切相关。AMI 指标同 ARI 指标一样，取值范围也是[-1,1]，值越大意味着聚类效果越好。

3.4　真实数据集实验及分析

为了验证算法在真实数据集上的效果，判断算法是否具有实际意义，分别采用 UCI 数据库中的 Iris、Wine、Vehicle 和 Landsat 共四个数据集进行实验。数据集详细信息如表 2 所示。

表 2　四种 UCI 数据

Table Four type of UCI data

数据集	实例数	维度	类别
Iris	150	4	3
Wine	178	13	3
Vehicle	846	18	4
Landsat	2000	36	6

经过真实数据集的实验后，ARI 指标和 AMI 指标的评价结果如表 3、4 所示。

表 3　各算法 ARI 指标对比

Table 3　ARI index of algorithms comparison

	KMEANS	NJW	STSC	DAN	NSGSC
Iris	0.7302	0.5638	0.8857	0.5609	0.9122
Wine	0.3711	0.3963	0.7987	0.3885	0.8035
Vehicle	0.0785	0.0254	0.1270	0.2765	0.3988
Landsat	0.2975	0.5564	0.5231	0.7231	0.7105

表 4　各算法 AMI 指标对比

Table 4　AMI index of algorithms comparison

	KMEANS	NJW	STSC	DAN	NSGSC
Iris	0.7484	0.5821	0.8623	0.5910	0.8968
Wine	0.4226	0.4371	0.7593	0.3755	0.7544
Vehicle	0.0923	0.0445	0.1555	0.2976	0.4135
Landsat	0.3234	0.5897	0.6222	0.7960	0.8142

由表 3 和 4 的对比分析可知，Iris 数据集上 ST-SC 算法和 NSG-SC 算法表现较好，具体在指标上 NSG-SC 算法略高于 ST-SC 算法；Wine 数据集上 ST-SC 算法和 NSG-SC 算法仍表现较好，其中 NSG-SC 算法的 ARI 指标高于 ST-SC 算法，AMI 指标低于 ST-SC 算法；Vehicle 数据集上所有算法表现均不理想，但 NSG-SC 算法的指标评价还是显著高于其他算法；Landsat 数据集上 DAN 算法和 NSG-SC 算法表现较好，其中 NSG-SC 算法的 ARI 指标低于 DAN 算法，AMI 指标高于 DAN 算法。综上所述，NSG-SC 算法在真实数据集上的表现依然优越，能够真实地反映出数据集的结构关系，从而得到更好的聚类结果。

4　结束语

本文提出了一种基于自然最近邻相似图的谱聚类算法。利用自然最近邻关系无须设定参数、能基于数据集自身特性进行搜索和受离散点影响小等优点，精确地划分出每个样本的邻域，构建相似图，从而得到能比传统谱聚类算法使用的 K 近邻法、ε 近邻法或全连接法更为真实地反映样本相似性关系的相似度矩阵，最后再进行谱聚类。在人工数据集和 UCI 真实数据集上

的实验表明 NSG-SC 算法在处理一些结构复杂的数据集时，能更好地反映出数据集的结构关系，从而得到更优秀的聚类结果。

算法还有提升空间，后续研究可考虑引入模糊近邻关系处理混合聚类或加入成对约束信息优化聚类效果，还可考虑与启发式算法结合。

参考文献：

[1] Xu Rui, Wunsch D. Survey of clustering algorithms [J]. IEEE Trans on Neural Networks, 2005, 16 (3): 645-678.

[2] Jain A K. Data clustering: 50 years beyond K-means [J]. Pattern Recognition Letters, 2010, 31 (8): 651-666.

[3] Park H S, Jun C H. A simple and fast algorithm for K-medoids clustering [J]. Expert Systems with Applications, 2009, 36 (2): 3336-3341.

[4] Yang Jing, Gao Jiawei, Liang Jiye, *et al.* An improved DBSCAN clustering algorithm based on data field [J]. Journal of Frontiers of Computer Science and Technology, 2012, 6 (10): 903-911.

[5] Kalita H K, Bhattacharyya D K, Kar A. A new algorithm for ordering of points to identify clustering Structure based on perimeter of triangle: OPTICS (BOPT) [C]// Proc of the 15th International Conference on Advanced Computing and Communications. Piscataway, NJ: IEEE Press, 2007: 523-528.

[6] Ansari S, Chetlur S, Prabhu S, *et al.* An overview if clustering analysis techniques used in data mining [J]. International Journal of Emerging Technology and Advanced Engineering, 2013, 3 (12): 284-246.

[7] Agarwal P, Alam M A, Biswas R. A hierarchical clustering algorithm for categorical attributes [C]// Proc of the 2nd International Conference on Computer Engineering and Applications. Piscataway, NJ: IEEE Press, 2010: 365-368.

[8] Zhou Yajian, Xu chen, Li Jiguo. Unsupervised anomaly detection method based on improved CURE clustering algorithm [J]. Journal on Communications, 2010, 31 (7): 18-23.

[9] Higham D J, Kalna G, Kibble M. Spectral clustering and its use in bioinformatics [J]. Journal of Computational and Applied Mathematics, 2007, 204 (1): 25-37.

[10] Chih-Hsuan W. Recognition of semiconductor defect patterns using spatial filtering and spectral clustering [J]. Expert Systems with Applications, 2008, 34 (3): 1914-1923.

[11] Shan Zeng, Rui Huang, Zhen Kang, *et al.* Image segmentation using spectral clustering of Gaussian mixture models [J]. Neurocomputing, 2014, 144: 346-356.

[12] He Ruifang, Qin Bing, Liu Ting. A novel approach to update summarization using evolutionary manifold-ranking and spectral clustering [J]. Expert Systems with Applications, 2012, 39 (3): 2375-2384.

[13] Ng A Y, Jordan M I, Weiss Y. On spectral clustering: analysis and an algorithm [C]// Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press, 2001: 849-856.

[14] Zelnik-Manor L, Perona P. Self-tuning spectral clustering [C]// Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press, 2004, 17: 1601-1608.

[15] Jia Hongjie, Ding Shifei, Xu Xinzheng, *et al.* The latest research progress on spectral clustering [J]. Neural Computing and Applications, 2014, 24: 1447-1486

[16] Yang Peng, Zhu Qingsheng, Huang Biao. Spectral clustering with density sensitive similarity function [J]. Knowledge-Based Systems, 2011, 24 (5): 621-628.

[17] Li Xinye, Guo Lijie. Constructing affinity matrix in spectral clustering based on neighbor propagation [J]. Neurocomputing, 2012, 97: 125-130.

[18] Blekas K, Lagaris I E. A spectral clustering approach based on Newton’s equations of motion [J]. International Journal of Intelligent Systems, 2013, 28 (4): 394-410.

[19] Inkaya T, Kayaligil S, Evin Ozdemirel N. An adaptive neighborhood construction algorithm based on density and connectivity [J]. Pattern Recognition Letters, 2015, 52: 17-24.

[20] Inkaya T. A parameter-free similarity graph for spectral clustering [J]. Expert Systems with Applications, 2015, 42: 9489-9498.

[21] Zhu Qingsheng, Feng Ji, Huang Jinlong. Natural neighbor: a self-adaptive neighborhood method without parameter K [J]. Pattern Recognition Letters, 2016, 80: 30-36.

[22] Fraley C, Raftery A E. Model-based clustering, discriminant analysis, and density estimation. [J] Journal of the American Statistical Association, 2002, 97: 611-631

[23] Still S, Bialek W. How many clusters?An information-theoretic perspective. [J] Neural Computation, 2004, 16 (12): 2483-2506.

[24] Santos J M, Embrechts M. On the use of the adjusted rand index as a metric for evaluating supervised classification [C]// Proc of International Conference on Artificial Neural Networks. Berlin: Springer, 2009: 175-184.

[25] Cai Deng, He Xiaofei, Han Jiawei. Document clustering using locality preserving indexing [J]. IEEE Trans on Knowledge and Data Engineering, 2005, 17 (12): 1624-1637.